

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25

2
3
4
5

6

7
8
9
10

12

13
14

- 15
16
17
18
19
20
21
22
23
24
25

1 In making out the rejection of this claim, the Office first argues that
2 Chapter 19 of Cohn teaches an application and Chapter 17 teaches an “application
3 interface” (API), as claimed, “comprising various types related to constructing
4 user interfaces”. However, the Office then argues that Chapters 5 and 6 teach “an
5 API comprising multiple types related to construction user interfaces” and
6 “classes which represent managed heap allocated data that has reference
7 assignment semantics”. The Office then relies on Flanagan for disclosing
8 “interfaces”, “structures” and “enumerations” and on Microsoft for disclosing
9 “delegates”. The Office states that it would have been obvious to integrate the
10 teachings of these references “because Visual J++ is based on Java, and Flanagan
11 teaches details of classes and packages offered by Java, and Microsoft teaches
12 delegates address many scenarios that are addressed by function pointers...”

13 Applicant respectfully disagrees and traverses the Office’s rejection. First,
14 Applicant submits that it remains unclear what teachings of Cohn the Office is
15 relying on for disclosing “an application program interface” (API), as claimed.
16 Specifically, on page 3 of the Office Action, the Office argues that Chapters 5 and
17 6 teaches an API, as claimed. Then, on page 6 of the Office Action, it
18 inexplicably argues that Chapter 19 teaches an API, as claimed.

19 Nevertheless, Applicant has thoroughly reviewed Chapters 5, 6, and 19
20 and is unable to find any discussion of an API, as recited in this claim.
21 Unfortunately, the Office offers no explanation other than to state: “[i]n Java, and
22 applets....Scrollbars; chapter 5, page 1 and Container class; chapter 6, page 1 and
23 through out chapters 5 and 6”. Applicant fails to see how this statement is
24 pertinent and submits that these excerpts simply do not teach the API recited in
25 this claim. Instead, Chapter 5 describes how to use Java components with respect

1 to creating a user interface and Chapter 6 describes how to combine and position
2 these components. Furthermore, Chapter 17 merely discusses the nature of URLs
3 and how they can be used. Missing from these excerpts, and from Cohn in
4 general, is any discussion of an API “comprising various types related to
5 constructing user interfaces”, as claimed. This is not surprising because Cohn is
6 described as a book that “tells how to use the language to solve problems” and
7 states: “[t]his book is not an API reference”. (See Introduction, page 12).

8 Additionally, Applicant respectfully submits that the Office is interpreting
9 the term “application program interface” in a manner that is inconstant with how
10 it is understood and used in the subject application. Applicant respectfully
11 reminds the Office that claims are not to be considered in a vacuum, but rather in
12 the context of the specification and drawings. (see e.g., MPEP 2106(II)(c),
13 2111(I)). In this regard, Applicant directs the Office’s attention to page 6 (lines
14 8-10) of the subject application, which is one example of how this term is
15 understood and used in the context of the subject application. Even a cursory
16 inspection of page 6 (lines 8-10) is sufficient to distinguish this claim from any
17 teachings found in the cited excerpts of Cohn. This excerpt is reproduced below
18 for the Office’s convenience:

19
20 As used herein, the phrase application program interface or API
21 includes traditional interfaces that employ method or function calls, as
22 well as remote calls (e.g., a proxy, stub relationship) and SOAP/XML
23 invocations.

24 Second, Applicant respectfully submits that the cited excerpts of Flanagan
25 neither disclose or suggest an API comprising all of the various types, as claimed.

1 For instance, applicant fails to see how Fig. 19-1 and page 238 of Flanagan
2 disclose “interfaces, that define a contract that other types can implement”, as
3 claimed. Instead, this figure and excerpt merely discuss *classes* of the java.awt
4 package.

5 Third, and perhaps most importantly, Applicant respectfully reminds the
6 Office that to support a conclusion that the claimed subject matter is directed to
7 obvious subject matter, either the references must expressly or impliedly suggest
8 the claimed subject matter or the examiner must present a convincing line of
9 reasoning as to *why* an artisan would have found the claimed subject matter to
10 have been obvious in light of the teachings of the references. (see MPEP 2142
11 and 2143.01).

12 Here, the Office’s only attempt at such an explanation is to state that it
13 would have been obvious to combine the teachings of these references “because
14 Visual J++ is based on Java, and Flanagan teaches details of classes and packages
15 offered by Java”. However, upon close examination, this statement merely
16 describes the references themselves and fails to explain *why* one would have been
17 motivated to combine their teachings. As such, this statement actually fails to
18 articulate any motivation at all. Accordingly, as best as Applicant can discern,
19 the Office’s argument relies solely on the fact that the references *can* be
20 combined. However, whether or not the references *can* be combined is irrelevant
21 because: “the mere fact that references can be combined or modified does not
22 render the resultant combination obvious unless the prior art also suggests the
23 desirability of the combination.” (MPEP 2143.01, citing *In re Mills*, 916 F.2d
24 680, 16 USPQ2d 1430 (Fed Cir. 1990)).
25

1 Furthermore, even if the cited references did disclose or suggest all the
2 claimed subject matter, which they do not, it remains unclear why one would
3 have been motivated to combine them in a manner similar to this claim. As noted
4 above, the Office has not provided any explanation, or even a stated motivation,
5 in this regard. Accordingly, applicant can only conclude that the Office has
6 impermissibly used the claimed invention as an instruction manual or 'template'
7 to piece together the teachings of the prior art so that the claimed invention is
8 rendered obvious. Applicant respectfully reminds the Office that "[o]ne cannot
9 use hindsight reconstruction to pick and choose among isolated disclosures in the
10 prior art to deprecate the claimed invention." (quoting *In Re Fine*, 837 F.2d 1071,
11 1075, 5 USPQ2d 1596, 1600 (Fed. Cir. 1988)).

12 Accordingly, in view of the above discussion, the Office has not
13 established a *prima facie* case of obviousness. Hence, for at least these reasons,
14 this claim is allowable.

15 **Claims 3-4** depend from claim 1 and are allowable as depending from an
16 allowable base claim. These claims are also allowable for their own recited
17 features which, in combination with those recited in claim 1, are neither disclosed
18 nor suggested in the references of record, either singly or in combination with one
19 another.

20 **Claim 5** recites an application program interface embodied on one or more
21 computer readable media, comprising: multiple types related to constructing user
22 interfaces, the types comprising classes which represent managed heap allocated
23 data that has reference assignment semantics, interfaces that define a contract that
24 other types can implement, delegates that are object oriented function pointers,
25

1 structures that represent static allocated data that has value assignment semantics
2 and enumerations which are value types that represent named constants.

3 In making out the rejection of this claim, the Office relies on the same
4 argument that it made with respect to claim 1 except that it does not rely on
5 Chapter 17 as teaching "an application program interface", as claimed. Therefore,
6 for the reasons set forth above, applicant respectfully traverses this rejection.

7 Accordingly, in view of the above discussion, the Office has not
8 established a *prima facie* case of obviousness. Hence, for at least these reasons,
9 this claim is allowable.

10 **Claims 6-15** depend from claim 5 and are allowable as depending from an
11 allowable base claim. These claims are also allowable for their own recited
12 features which, in combination with those recited in claim 5, are neither disclosed
13 nor suggested in the references of record, either singly or in combination with one
14 another.

15 Additionally, regarding claims 8-15, Applicant submits that the excerpts
16 from Cohn that are cited by the Office disclose various *classes*, not interfaces.
17 Accordingly, the Office's reliance on these excerpts is misplaced.

18 **Claim 16** recites a distributed computer software architecture comprising:

- 19
- 20 • one or more applications configured to be executed on one or more
21 computing devices, the applications handling requests submitted
22 from remote computing devices;
 - 23 • a networking platform to support the one or more applications; and
 - 24 • an application programming interface to interface the one or more
25 applications with the networking platform, the application
programming interface comprising various types related to
constructing user interfaces, wherein the various types comprise:
 - classes which represent managed heap allocated data that has
reference assignment semantics;

- interfaces that define a contract that other types can implement;
- delegates that are object oriented function pointers;
- structures that represent static allocated data that has value assignment semantics; and
- enumerations which are value types that represent named constants.

In making out the rejection of this claim, the Office relies on the same argument that it made with respect to claim 1. Therefore, for the reasons set forth above, applicant respectfully traverses this rejection.

Accordingly, in view of the above discussion, the Office has not established a *prima facie* case of obviousness. Hence, for at least these reasons, this claim is allowable.

Claims 18-27 depend from claim 16 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which, in combination with those recited in claim 16, are neither disclosed nor suggested in the references of record, either singly or in combination with one another.

Additionally, regarding claims 20-27, Applicant submits that the excerpts from Cohn that are cited by the Office disclose various classes, not interfaces. Accordingly, the Office's reliance on these excerpts is misplaced.

Claim 28 recites a computer system including one or more microprocessors and one or more software programs, the one or more software programs utilizing an application program interface to request services from an operating system, the application program interface including separate commands to request services comprising services related to constructing user interfaces, wherein the application program interface groups API functions into multiple namespaces that define a collection of classes which represent managed heap

1 allocated data that has reference assignment semantics, interfaces that define a
2 contract that other types can implement, delegates that are object oriented
3 function pointers, enumerations which are value types that represent named
4 constants and structures that represent static allocated data that has value
5 assignment semantics.

6 In making out the rejection of this claim, the Office relies on the same
7 argument that it made with respect to claim 1. Therefore, for the reasons set forth
8 above, applicant respectfully traverses this rejection.

9 Accordingly, in view of the above discussion, the Office has not
10 established a *prima facie* case of obviousness. Hence, for at least these reasons,
11 this claim is allowable.

12 **Claim 29** recites a method comprising:

- 13
- 14 • managing network and computing resources for a distributed
computing system; and
 - 15 • exposing a set of functions that enable developers to access the
16 network and computing resources of the distributed computing
17 system, the set of functions comprising functions to facilitate
18 construction of user interfaces, wherein the functions are grouped
19 into multiple namespaces that define a collection of classes which
20 represent managed heap allocated data that has reference
21 assignment semantics, interfaces that define a contract that other
types can implement, delegates that are object oriented function
pointers, enumerations which are value types that represent named
constants and structures that represent static allocated data that has
value assignment semantics.

22 In making out the rejection of this claim, the Office simply indicates “see
23 rejection of claim 1 above.” Therefore, for the reasons set forth above, applicant
24 respectfully traverses this rejection. In addition, applicant respectfully submits
25

1 that the cited references do not disclose or suggest "exposing a set of functions",
2 as claimed.

3 Accordingly, in view of the above discussion, the Office has not
4 established a *prima facie* case of obviousness. Hence, for at least these reasons,
5 this claim is allowable.

6 **Claim 30** depends from claim 29 and is allowable as depending from an
7 allowable base claim. This claim is also allowable for its own recited features
8 which, in combination with those recited in claim 30, are neither disclosed nor
9 suggested in the references of record, either singly or in combination with one
10 another.

11 **Claim 31** recites a method, comprising creating a namespace with
12 functions that enable drawing and construction of user interfaces, the name space
13 defining classes which represent managed heap allocated data that has reference
14 assignment semantics, interfaces that define a contract that other types can
15 implement, delegates that are object oriented function pointers, structures that
16 represent static allocated data that has value assignment semantics, and
17 enumerations which are value types that represent named constants.

18 In making out the rejection of this claim, the Office simply indicates "see
19 rejection of claim 5 above." Therefore, for the reasons set forth above, applicant
20 respectfully traverses this rejection. In addition, applicant respectfully submits
21 that the cited references do not disclose or suggest "creating a namespace", as
22 claimed.

23 Accordingly, in view of the above discussion, the Office has not
24 established a *prima facie* case of obviousness. Hence, for at least these reasons,
25 this claim is allowable.

1 **Claims 32-40** depend from claim 31 and are allowable as depending from
2 an allowable base claim. These claims are also allowable for their own recited
3 features which, in combination with those recited in claim 31, are neither
4 disclosed nor suggested in the references of record, either singly or in
5 combination with one another.

6 Additionally, regarding claims 34-40, Applicant submits that the excerpts
7 from Cohn that are cited by the Office disclose various classes, not interfaces.
8 Accordingly, the Office's reliance on these excerpts is misplaced

9
10 **Conclusion**

11 Applicant respectfully submits that the Office has failed to establish a
12 *prima facie* case of obviousness for the reasons set forth above. Applicant
13 respectfully requests a Notice of Allowability be issued forthwith.

14
15
16 Respectfully submitted,

17
18 Dated: 6/19/2006

19 By: 

20 Rich Bucher
21 Reg. No. 57,971
22 (509) 324-9256 ext. 216
23
24
25